Obtaining Optimal PIDA Controller for Temperature Control of Electric Furnace System via Flower Pollination Algorithm

Noppadol PRINGSAKUL and Deacha PUANGDOWNREONG*
Department of Electrical Engineering
Graduate School, Southeast Asia University
19/1 Petchakasem Rd., Nongkhaem, Bangkok, 10160
THAILAND
*corresponding author :deachap@sau.ac.th   http//:www.sau.ac.th

Chaiyo THAMMARAT and Sarot HLANGNAMTHIP
Department of Electrical Engineering
Faculty of Engineering, Southeast Asia University
19/1 Petchakasem Rd., Nongkhaem, Bangkok, 10160
THAILAND

Abstract:- The electric furnace temperature control system is one of the real-world second-order systems plus time delay (SOSPD) widely used as the process control in industries. It is normally operated under the PID feedback control loop. The PIDA controller, however, performed better response than the PID controller for higher order plant. In this paper, an optimal PIDA controller design for the electric furnace temperature control system is presented. Regarding to modern optimization context, the flower pollination algorithm (FPA) which is one of the most efficient population-based metaheuristic optimization techniques is applied to search for the appropriate PIDA's parameters. The proposed FPA-based PIDA design framework is considered as the constrained optimization problem. System responses obtained by the PIDA controller designed by the FPA will be compared with those obtained by the PID controller also designed by the FPA. As results, it was found that the PIDA can provides the very satisfactory tracking and regulating responses of the electric furnace temperature control system superior to the PID, significantly.

Key-Words: - PIDA Controller, Electric Furnace Temperature Control, Metaheuristic Optimization

1 Introduction

The electric furnace temperature control system is one of the real-world second-order systems plus time delay (SOSPD) [1] widely used in various industrial production processes [1],[2]. The electric furnace uses electricity as its main power source in order to generate heat for industrial uses. By literatures, temperatures of the electric furnace system can be effectively controlled by the PID controller [3], fuzzy controller [4], fuzzy-PID controller [5] and neural network [6]. The PID controller is most popular and easiest way for the electric furnace temperature control. Based on the modern optimization context, the PID design problem for the electric furnace system can be considered as one of the constrained optimization problems that can be efficiently solved by metaheuristics, for example, the design of PID controller for the electric furnace temperature system by Nelder-Mead (NM) algorithm [7], by Genetic Algorithm (GA) [8],[9] and by Flower Pollination Algorithm (FPA) [10].

However, the PIDA controller, possessing three arbitrary zeros and one pole at origin, performed faster and smoother responses for the higher-order plants than the PID controller. The PIDA controller, firstly proposed by Jung and Dorf in 1996 [11], was conducted and designed by several metaheuristics, such as GA [12], Particle Swarm Optimization (PSO) [13],[14], Current Search (CuS) [15], Firefly Algorithm (FA) [16], Bat Algorithm (BA) [17] and Cuckoo Search (CS) [18].

In 2012, the FPA was firstly proposed by Yang [19] as one of the most powerful population-based metaheuristics for solving the optimization problems. The FPA algorithm mimics the behavior of pollination of flowering plant in nature associated with the Lévy flight distribution in order to generate the elite solutions. By literatures, the FPA was applied to solve many real-world optimization problems, for example, pressure vessels design [19], disc break design [20], traveling transportation problem [21], control system design [10],[22],[23],[24] and model identification [25]. In this paper, the FPA is applied to obtain an optimal PIDA controller for the electric furnace temperature control system. Results obtained by the PIDA controller designed by the FPA will be compared with those obtained by the PID designed by the FPA. The rest of the paper is provided as follows. The PIDA control loop for the electric furnace temperature system is given in section 2. Problem formulation consisting of FPA algorithms and FPA-based PIDA controller design for the electric furnace temperature control system is described in section 3. Results and discussions are illustrated in section 4. Conclusions are followed in section 5.

2 PIDA Control Loop

The electric furnace temperature system operated under the PIDA feedback control loop is represented by the block diagram as shown in Fig. 1.

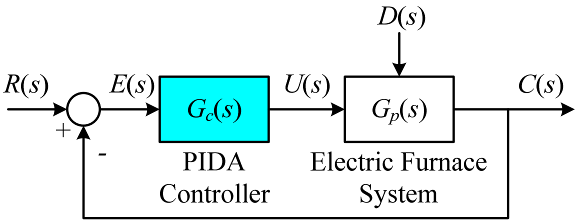


Fig. 1 PIDA feedback control loop.

2.1 Electric Furnace Temperature System

Fig. 2 shows the schematic diagram of the electric furnace temperature system [26] consists of electrical furnace, controller, thermocouple and heater in order to control the temperature in electrical furnace. Referring to Fig. 2, r is input voltage, U is output voltage from controller, y is output voltage from thermocouple and R is armature resistance.

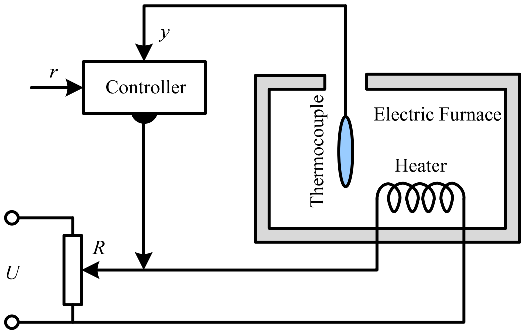


Fig. 2 Electric furnace temperature system [26].

The s-domain transfer function of the electric furnace temperature system $G_p(s)$ was formulated as the second-order system plus time delay (SOSPD) [26] as given in (1). The time delay (or transport lag) in (1) is approximated by (2). Then, the s-domain transfer function of the electric furnace temperature system in (1) can be rewritten as expressed in (3). The model $G_p(s)$ in (3) will be used as the plant in the feedback control loop shown in Fig. 1.

$$G_p(s) = \frac{0.15}{s^2 + 1.1s + 0.2} e^{-1.5s} \tag{1}$$

$$e^{-1.5s} = \frac{1 - 0.75s}{1 + 0.75s} \tag{2}$$

$$G_p(s) = \frac{-0.1125s + 0.15}{0.75s^3 + 1.825s^2 + 1.25s + 0.2} \tag{3}$$

2.2 PIDA Controller

Proposed by Jung and Dorf in 1996 [11], the s-domain transfer function of the PIDA controller is stated in (4), where K_p, K_i, K_d and K_a are proportional, integral, derivative and accelerated gains, respectively. Referring to (4), a, b, c and d, e are zeros and poles of the PIDA controller. Once a, b, c << d, e, the poles d, e can be neglected [11]. Due to this, the PIDA transfer function in (4) can be rewritten as expressed in (5). It can be observed that the PIDA controller possesses three arbitrary zeros and one pole at origin. Regarding to Fig. 1, the PIDA controller $G_c(s)$ receives the error signal $E(s)$ and produces the control signal $U(s)$ to control the output response $C(s)$ and regulate the external disturbance signal $D(s)$ referring to the reference input $R(s)$.

$$\left.\begin{aligned} G_c(s)\big|_{PIDA} &= K_p + \frac{K_i}{s} + \frac{K_d s}{(s+d)} + \frac{K_a s^2}{(s+d)(s+e)} \\ &= \frac{K(s+a)(s+b)(s+c)}{s(s+d)(s+e)} \end{aligned}\right\} \tag{4}$$

$$\left.\begin{aligned} G_c(s)\big|_{PIDA} &= K_p + \frac{K_i}{s} + K_d s + K_a s^2 \\ &= \frac{K_a s^3 + K_d s^2 + K_p s + K_i}{s} \end{aligned}\right\} \tag{5}$$

3 Problem Formulation

In this section, the problem formulation is presented. The FPA algorithms are briefly described and the FPA-based PIDA controller design for the electric furnace temperature control system is then given as follows.

3.1 FPA Algorithms

In nature, the objective of the flower pollination is the survival of the fittest and optimal reproduction of flowering plants. Pollination in flowering plants can take two major forms, i.e. biotic and abiotic [27]. About 90% of flowering plants belong to biotic pollination. Pollen is transferred by a pollinator such as bees, birds, insects and animals. About 10% remaining of pollination takes abiotic such as wind and diffusion in water. Pollination can be achieved by self-pollination or cross-pollination as visualized in Fig. 3 [28],[29]. Self-pollination is the fertilization of one flower from pollen of the same flower (Autogamy) or different flowers of the same plant (Geitonogamy). They occur when a flower contains both male and female gametes. Self-pollination usually occurs at short distance without pollinators. It is regarded as the local pollination. Cross-pollination, Allogamy, occurs when pollen grains are moved to a flower from another plant. The process happens with the help of biotic or abiotic agents as pollinators. Biotic, cross-pollination may occur at long distance with biotic pollinators. It is regarded as the global pollination. Bees and birds as biotic pollinators behave Lévy flight behaviour [30] with jump or fly distance steps obeying a Lévy distribution. The FPA algorithm proposed by Yang [19] can be summarized by the pseudo code as visualized in Fig. 4.

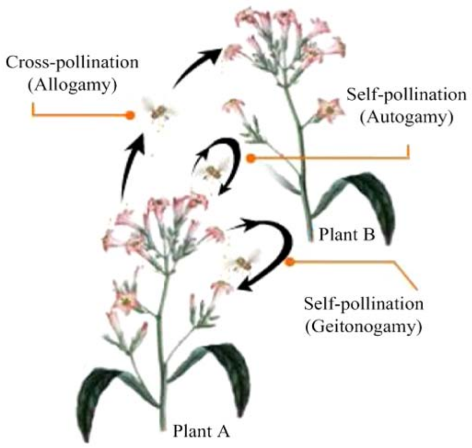
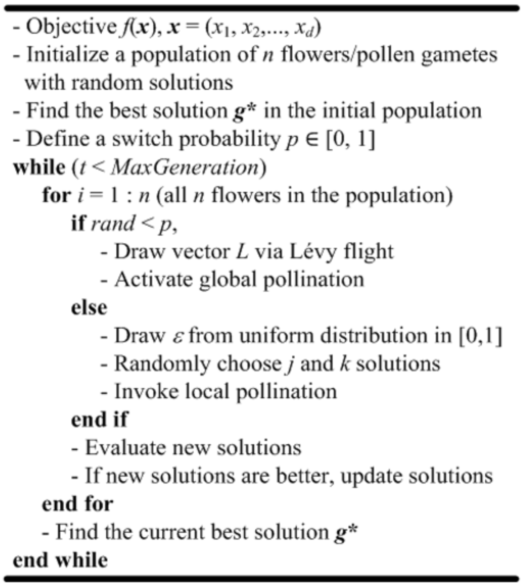


Fig. 3 Flower pollination in nature [28],[29].



```
- Objective $f(\mathbf{x})$, $\mathbf{x} = (x_1, x_2,..., x_d)$
- Initialize a population of $n$ flowers/pollen gametes
  with random solutions
- Find the best solution $\mathbf{g}^*$ in the initial population
- Define a switch probability $p \in [0, 1]$
while ($t < MaxGeneration$)
   for $i = 1 : n$ (all $n$ flowers in the population)
      if $rand < p$,
         - Draw vector $L$ via Lévy flight
         - Activate global pollination
      else
         - Draw $\varepsilon$ from uniform distribution in [0,1]
         - Randomly choose $j$ and $k$ solutions
         - Invoke local pollination
      end if
      - Evaluate new solutions
      - If new solutions are better, update solutions
   end for
   - Find the current best solution $\mathbf{g}^*$
end while
```

Fig. 4 Pseudo code of FPA algorithms [19].

In FPA algorithms, a solution $\mathbf{x}_i$ is equivalent to a flower and/or a pollen gamete. For global pollination, flower pollens are carried by pollinators. With Lévy flight, pollens can travel over a long distance as expressed in (6), where $\mathbf{g}^*$ is the current best solution found among all solutions at the current generation/iteration t, and L stands for the Lévy flight that can be approximated by (7), while $\Gamma(\lambda)$ is the standard gamma function as given in (8). The local pollination can be represented by (9), where $\mathbf{x}_j$ and $\mathbf{x}_k$ are pollens from the different flowers of the same plant species, while ε stands for random walk by using uniform distribution, where $\varepsilon \in [0,1]$. Flower pollination activities can occur at all scales, both local and global pollination. In this case, a switch probability or proximity probability p is used to switch between global pollination and local pollination. From Yang's recommendations [19], the number of pollens $n = 25$, proximity probability $p = 0.8$ and $\lambda = 1.5$ works better for most applications. Therefore, these recommendations are then conducted in this work.

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + L(\mathbf{x}_i^t - \mathbf{g}^*) \tag{6}$$

$$L \approx \frac{\lambda\Gamma(\lambda)\sin(\pi\lambda/2)}{\pi}\frac{1}{s^{1+\lambda}} \tag{7}$$

$$\Gamma(\lambda) = \int_0^\infty x^{\lambda-1}e^{-x}dx \tag{8}$$

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \varepsilon(\mathbf{x}_j^t - \mathbf{x}_k^t) \tag{9}$$

3.2 FPA-Based PIDA Design

The FPA-based PIDA controller design for the electric furnace temperature control system can be represented by the block diagram as shown in Fig. 5. The objective function J is performed as sum-squared error between the reference temperature $R(s)$ and the actual temperature $C(s)$ stated in (10). J will be sent to the FPA block to be minimized by searching for the optimal values of K_p, K_i, K_d and K_a as the parameters of the PIDA controller within their particular boundaries or search spaces $K \in [K_{min}, K_{max}]$ as shown in (11). In this work, J will be minimized according to the constrained functions as defined in (12), where t_r and t_{r_max} are rise time and maximum rise time, M_p and M_{p_max} are percent overshoot and maximum percent overshoot, t_s and t_{s_max} are settling time and maximum settling time, and E_{ss} and E_{ss_max} are steady-state error and maximum steady-state error, respectively.

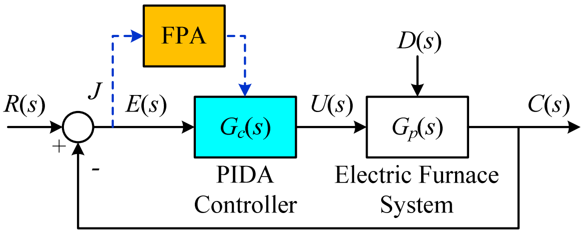


Fig. 5 FPA-based PIDA controller design.

$$Minimize \quad J = \sum_{i=1}^{N} (R_i - C_i)^2 \tag{10}$$

$$\left.\begin{aligned} Subject\ to\quad & t_r \le t_{r\_max}, \quad M_p \le M_{p\_max}, \\ & t_s \le t_{s\_max}, \quad E_{ss} \le E_{ss\_max}, \\ & K_{p\_min} \le K_p \le K_{p\_max}, \\ & K_{i\_min} \le K_i \le K_{i\_max}, \\ & K_{d\_min} \le K_d \le K_{d\_max}, \\ & K_{a\_min} \le K_a \le K_{a\_max}, \end{aligned}\right\} \tag{11}$$

4 Results and Discussions

In order to apply the FPA for designing the optimal PIDA controller for the electric furnace temperature control system, the FPA algorithms were coded by MATLAB version 2017b (License No.#40637337) run on Intel(R) Core(TM) i5-3470 CPU@3.60GHz, 4.0GB-RAM. The FPA's parameters, i.e. number of flower (pollen gametes) n = 40 and a probability p = 0.2 (20%) for switching between local pollination and global pollination, are set according to recommendations of Yang [19]. Search spaces and constraint functions in (11) are then performed as given in (12). The maximum generation *MaxGeneration* = 500 is then set as the termination criteria (TC). 50 trials are conducted to find the optimal PIDA controller for the electric furnace temperature control system. For comparison with the PID controller, K_a in (5) will be set as zero.

$$\left.\begin{aligned} Subject\ to\quad & t_r \le 4.5\,\text{sec.}, \quad M_p \le 10.0\,\%, \\ & t_s \le 10.0\,\text{sec.}, \quad E_{ss} \le 0.01\,\%, \\ & 0 \le K_p \le 5.0, \quad 0 \le K_i \le 1.0, \\ & 0 \le K_d \le 5.0, \quad 0 \le K_a \le 1.0 \end{aligned}\right\} \tag{12}$$

When 50 trials of the search process were completed, the FPA can successfully provide the optimal PID and PIDA controllers for the electric furnace temperature control system as shown in (13) and (14), respectively.

$$G_c(s)\big|_{PID} = 3.55 + \frac{0.61}{s} + 3.85s \tag{13}$$

$$G_c(s)\big|_{PIDA} = 3.98 + \frac{0.66}{s} + 4.99s + 0.99s^2 \tag{14}$$

The convergent rates of the objective functions in (10) associated with inequality constraint functions in (12) proceeded by the FPA over 50 trials are plotted in Fig. 6.

The step-input (or tracking) responses of the electric furnace temperature control system without controller, with PID controller and with PIDA controller designed by the FPA are depicted in Fig. 7. Referring to Fig. 7, the electric furnace temperature system without controller provides t_r = 14.56 sec., t_s = 20.12 sec., without M_p and E_{ss} = 0.2520 (25.20%). With the PID controller, the step response of the electric furnace temperature control system yields t_r = 3.61 sec., t_s = 8.46 sec., M_p = 8.95% and without E_{ss}. Finally, the step response of the electric furnace temperature control system with the PIDA controller gives t_r = 3.61 sec., t_s = 5.18 sec., M_p = 2.95% and without E_{ss}.

The step-disturbance (or regulating) responses of the electric furnace temperature control system with PID controller and with PIDA controller designed by the FPA are plotted in Fig. 8. From such the figure, the electric furnace temperature control system with PID controller provides the maximum overshoot from regulating M_{p_reg} = 22.75% and recovering time from regulating t_{r_reg} =

16.78 sec. However, the step-disturbance response of the electric furnace temperature control system with the PIDA controller gives M_{p_reg} = 20.01% and t_{r_reg} = 16.49 sec. This can be noticed that the PIDA controller designed by the FPA can provide very satisfactory response of the electric furnace temperature control system superior to the PID controller.

5 Conclusions

In this paper, the application of the flower pollination algorithm (FPA) to design an optimal PIDA controller for the electric furnace temperature control system has been proposed. The electric furnace temperature system has been conducted in this work as one of the real-world second-order systems plus time delay (SOSPD) which were widely used in industries. Based on modern optimization, the FPA, one of the most efficient population-based metaheuristic optimization techniques, has been applied. Simulation results have shown that the PID and PIDA controllers designed by the FPA can provide vary satisfactory response of the electric furnace temperature control system according to the predefined constrained functions. However, the PIDA could provide the very satisfactory step-input (or tracking) and step-disturbance (or regulating) responses of the electric furnace temperature control system better than the PID controller.

References:

[1] A. Dwyer, *Handbook of PI and PID Controller Tuning Rules*, Imperial College, 2009.

[2] B. G. Liptak, *Instrumentation Engineer's Handbook, Process Control*, CRC Press, 1995.

[3] Y. Han, J. Jinling, C. Guangjian and C. Xizhen, Temperature Control of Electric Furnace based on Fuzzy PID, *IEEE Trans. ICEOE*, 2011, pp. V3-41-V3-44.

[4] F. Teng and H. Li, Adaptive Fuzzy Control for the Electric Furnace, *IEEE Trans. ICIS*, 2009, pp. 439-443.

[5] X. Junming, Z. Haiming, J. Lingyun and Z. Rui, Based on Fuzzy-PID Self-Tuning Temperature Control System of the Furnace, *IEEE Trans. ICEICE*, 2011, pp. 15-17.

[6] W. Ding-du, 2010. Decoupling Control of Electric Heating Furnace Temperature Based on DRNN Neural Network, *IEEE Trans. ICECT*, 2010, pp. 261-264.

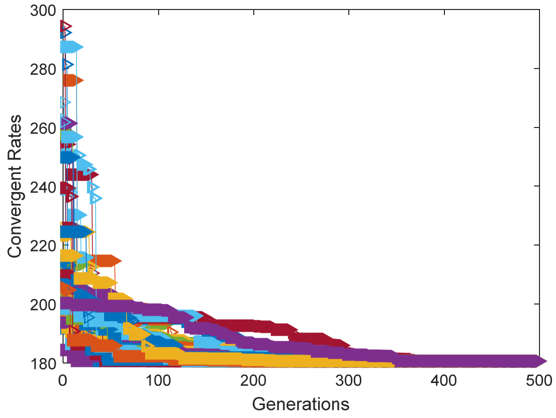


Fig. 6 Convergent rates over 50 trials.

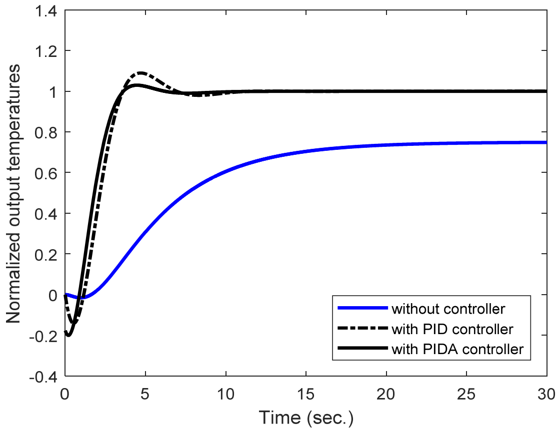


Fig. 7 Step-input (tracking) responses of the electric furnace temperature control system.

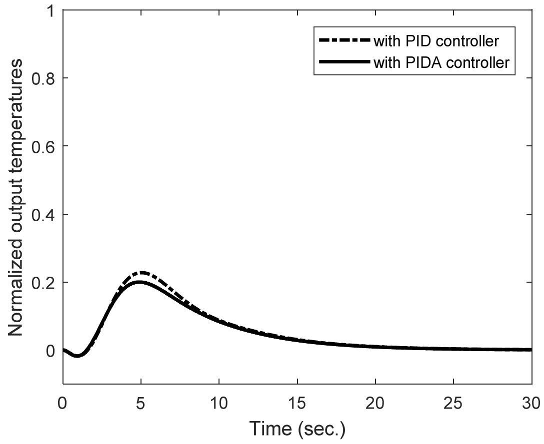


Fig. 8 Step-disturbance (regulating) responses of the electric furnace temperature control system.

[7] V. Sinlapakun and W. Assawinchaichote, PID Controller Based Nelder Mead Algorithm for Electric Furnace System with Disturbance, *ECTI Trans. Computer and Information Technology*, Vol. 10, No. 1, 2016, pp. 71-79.

[8] B. Srilakshmi and K. Venkataratnam, Temperature Control of Electric Furnace using Genetic Algorithm Based PID Controller, *International Journal of Advanced Engineering and Global Technology*, Vol. 3, No. 11, 2015, pp. 1348-1352.

[9] D. Sain, S. K. Swain, S. K. Mishra and S. Dutta, Robust Set-Point Weighted PID Controller Design Using Genetic Algorithm for Electric Furnace Temperature Control System, *International Journal of Control Theory and Applications*, Vol. 9, No. 39, 2016, pp. 29-36.

[10] S. Hlungnamthip, N. Pringsakul, A. Nawikavatan and D. Puangdownreong, FPA-Based PID Controller Design for Temperature Control of Electric Furnace System, *The 2018 International Conference on Engineering and Natural Science (ICENS 2018)*, 2018, pp. 60-68.

[11] S. Jung and R. C. Dorf, Analytic PIDA Controller Design Technique for a Third Order System, *Proceedings of the* 35*th IEEE Conference on Decision and Control*, Kobe, 1996, pp. 2513-2518.

[12] S. Sornmuang and S. Sujitjorn, GA-Based PIDA Control Design Optimization with an Application to AC Motor Speed Control, *International Journal of Mathematics and Computers in Simulation*, Vol. 4, No. 3, 2010, pp. 67-80.

[13] D. Puangdownreong and S. Suwannarongsri, Torsional Resonance Suppression via PIDA Controller Designed by the Particle Swarm Optimization, *The ECTI–CON International Conference*, 2008, pp. 673-676.

[14] C. Thammarat, P. Sukseam, D. Puangdownreong and S. Suwannarongsri, Design of PIDA Controllers via Particle Swarm Optimization, *The ANSCSE 2008 Symposium*, 2008, pp. 393-398.

[15] D. Puangdownreong, Application of Current Search to Optimum PIDA Controller Design, *Intelligent Control and Automation*, Vol .3, No.4, 2012, pp. 303-312.

[16] D. Puangdownreong, S. Sumpunsri, M. Sukchum, C. Thammarat, S. Hlangnamthip and A. Nawikavatan, FA-Based Optimal PIDA Controller Design for AVR System, *The iEECON2018 International Conference*, 2018, pp. 548-551.

[17] C. Thammarat, K. Lurang, D. Puangdownreong, S. Suwammarongsri, S. Hlangnamthip and A. Nawikavatan, Application of Bat-Inspired Algorithm to Optimal PIDA Controller Design for Liquid-Level System, *The iEECON2018 International Conference*, 2018, pp. 556-559.

[18] A. Nawikavatan, T. Jitwang, C. Thammarat and D. Puangdownreong, Application of Cuckoo Search to Optimal PIDA Controller Design for Three-Tank Liquid-Level Control System, *The 2018 International Conference on Engineering and Natural Science (ICENS 2018)*, 2018, pp. 51-59.

[19] X. S. Yang, Flower Pollination Algorithm for Global Optimization, *Unconventional Computation and Natural Computation*, *Lecture Notes in Computer Science*, Vol. 7445, 2012, pp. 240-249.

[20] X. S. Yang, M. Karamanoglua and X. He, Multi-Objective Flower Pollination Algorithm for Optimization, *Procedia Computer Science*, Vol. 18, 2013, pp. 861-868.

[21] S. Suwannarongsri and D. Puangdownreong, Application of Flower Pollination Algorithm for Solving Medium Scaled Traveling Transportation Problems, *The 2018 International Conference on Engineering and Natural Science (ICENS 2018)*, 2018, pp. 34-42.

[22] D. Puangdownreong, Optimal State-Feedback Design for Inverted Pendulum System by Flower Pollination Algorithm, *International Review of Automatic Control (IREACO)*, Vol. 9, No. 5, 2016, pp. 289-297.

[23] C. Thammarat, A. Nawikavatan and D. Puangdownreong, Application of Flower Pollination Algorithm to PID Controller Design for Three-Tank Liquid-Level Control System, *The 9th International Conference on Sciences, Technology and Innovation for Sustainable Well-Being (STIWB)*, 2017, pp. 42-46.

[24] S. Hlungnamtip, C. Thammarat and D. Puangdownreong, Obtaining Optimal PID Controller for DC Motor Speed Control System via Flower Pollination Algorithm, *The 9th International Conference on Sciences, Technology and Innovation for Sustainable Well-Being (STIWB)*, 2017, pp. 52-56.

[25] D. Puangdownreong, C. Thammarat, S. Hlungnamtip and A. Nawikavatan, Application of Flower Pollination Algorithm to Parameter Identification of DC Motor Model, *The 2017 International Electrical Engineering Congress (iEECON–2017)*, Vol. 2, 2017, pp. 711-714.

[26] J. Paulusov and M. Dbravsk, Application of Design of PID Controller for Continuous Systems, *FEI STU*, Slovak Republic, 2012.
[27] B. J. Glover, *Understanding Flowers and Flowering: An Integrated Approach*, Oxford University Press, 2007.
[28] P. Willmer, *Pollination and Floral Ecology*, Princeton University Press, 2011.
[29] K. Balasubramani and K. Marcus, A Study on Flower Pollination Algorithm and Its Applications, *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, Vol. 3, 2014, pp. 320-325.
[30] I. Pavlyukevich, Lévy Flights, Non-Local Search and Simulated Annealing, *Journal of Computational Physics*, Vol. 226, 2007, pp. 1830-1844.